

(19) Japan Patent
Office (JP)

(12) **Unexamined Patent Digest** (A)

(11) Unexamined Patent
Application Publication
Number
2002-63102
(P2002-63102A)

(43) Publication date: February 28, 2002

(51) Int. Cl. ⁴	Identification codes	FI	Theme Codes (Reference)
G 06 F 13/00	540	G 06 F 13/00	540R

Request for examination: Filed Number of claims: 13 OL (18 pages total)

(21) Application number	2000-247126 (P2000- 247126)	(71) Applicant	300064847 Frontline Dot JP, Ltd. Alco Tower 5F, 8-1 Shimo-Meguro 1-chome, Meguro-ku, Tokyo Japan
(22) Date of application	August 16, 2000	(72) Inventor	Saito Takashi c/o Frontline Dot JP, Ltd. Alco Tower 5F, 8-1 Shimo-Meguro 1-chome, Meguro-ku, Tokyo Japan
		(74) Agent	Nomura , Patent Agent

(54) Title of the Invention: Log analysis system

[Abstract]

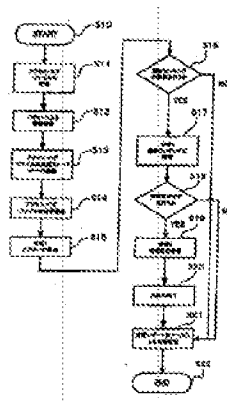
[Problem]

To calculate the number of estimated visitors from access logs, and to easily calculate page transition rankings.

[Means of Solving the Problem]

The present invention is a system for identifying estimated visitors based on IP addresses, the type of Web browser, and the operating system under which that Web browser runs, and for analyzing the number of visitors by positing the number of people who continuously access a site for a definite length of time as estimated visitors. In addition, a robot search engine is used to search for benchmark pages that either have had a META tag embedded or have been designated as benchmark Web pages by a site administrator. Page transitions are observed by pivoting off these pages, and the number of visitors who navigated to specified pages is calculated. Also, log data from a plurality of servers is efficiently analyzed by combining these means.

[Selected drawing: Fig. 2]



LOG ANALYSIS SYSTEM

What Is Claimed Is:

1. A log analysis system for obtaining an access log for a prescribed Web server and dynamically analyzing access trends to the Web server by users based on that log file, wherein the number of visitors is estimated on the basis of IP addresses, identity of Web browser and operating system type, and the time interval between HTML file session transitions.
2. The log analysis system according to Claim 1, wherein in a case in which the time interval between session transitions is 30 minutes or less, a determination is made that the same visitor [has accessed the server].
3. The log analysis system according to Claim 1 or Claim 2, wherein the Web browser and operating system type comprises version information.
4. The log analysis system according to any of Claim 1 through Claim 3, wherein access trends to a Web server by users are dynamically analyzed by specifying a specific HTML file, locating that specific HTML file based on the log file using robot search means, and analyzing the chronological transitions between the HTML files before and after the specific HTML file.
5. The log analysis system according to Claim 4, wherein the specific HTML file is specified by embedding a specific META tag in the header section of the specific HTML file in advance and locating the META tag based on the log file using robot search means.
6. The log analysis system according to Claim 4 or Claim 5, wherein logs having a specific pattern of chronological transitions between the HTML files are aggregated.
7. The log analysis system according to Claim 4 or Claim 5, wherein logs partially matching with a specific pattern for chronological transitions between the HTML files are aggregated.

8. The log analysis system according to Claim 1, wherein a plurality of prescribed Web servers exists, and wherein the access logs for the plurality of Web servers are converted to a common format and transferred to a log database on a log analysis site.
9. The log analysis system according to Claim 8, wherein the plurality of Web servers comprises servers of different types, and server names along with access logs are transferred to the log database.
10. The log analysis system according to Claim 8, wherein the plurality of Web servers comprises servers of the same type, and only access logs are transferred to the log database.
11. The log analysis system according to any of Claims 1-9 {*1}, wherein a report is prepared showing the number of accesses, the number of visitors, and the access time on the basis of the estimated number of visitors.
12. The log analysis system according to Claim 11, wherein a report is prepared showing parameter-specific rankings of any parameter based on the number of accesses, the number of visitors, and the access time.
13. The log analysis system according to Claim 11 or Claim 12, wherein a report is prepared also showing the title, obtained by robot search means, in the header section of each analyzed HTML file.

SUMMARY OF THE INVENTION

[0001]

1. Field of the Invention

This invention relates to an administration system for managing and analyzing access logs over the Internet.

[0002]

2. Background of the Invention (Prior Art)

Conventionally, various systems have been proposed as systems for analyzing visitors on the Internet.

[0003]

In order to measure or estimate the number of visitors, some systems use a cookie ID provided in browsers that are required for Internet communications, and other systems use IP addresses. In addition, some systems perform analyses by obtaining all access log, analyzing the logs statistically and matching log records with visitor IDs in order to determine how visitors are making page visits on sites on the Internet. Also, in order to efficiently process access logs, systems have adopted mechanisms for efficiently tracking visitor numbers and tabulating Web site page visits (page transitions) by installing hardware known as a collector in front of the router in order to collect only logs at the packet level. Moreover, robot search engines are often used in searching for Web pages on the Internet.

[0004]

However, if viewed in the context of using cookie IDs, for example, for estimating the number of visitors, there is a disadvantage in that attempts to accurately measure the number of visitors observes only those people who have authorized cookies on their systems, since some users block issuance of cookies because of privacy concerns. Also, many servers lack the capability to embed cookies, and in this case it is impossible to collect access logs using cookies.

[0005]

In addition, because when traffic and transactions pass through a firewall, the IP address becomes the IP address of the firewall, there is a significant possibility when using IP addresses for analysis that accesses will be interpreted as having been performed by a single person even though in fact a great many have actually accessed the server, especially in the case of accesses from corporations.

[0006]

In addition, since analyzing all logs involves handling a staggering amount of data, considerable time and trouble is involved in aggregation and analysis, imposing massive costs in terms of both computer resources and money, especially when analyzing page transitions.

[0007]

Finally, once a plurality of Web servers is involved, consolidation of access log data becomes even more complicated, and very great amounts of time and trouble are involved when consolidating logs. A solution is needed to resolve these issues efficiently.

[0008]

[Problem the Invention Is to Solve]

An object of the present invention is to provide a method for simply and accurately identifying when the same visitor is accessing a site, and to provide a method capable of easily tracking page transitions by these visitors.

[0009]

[Means of Solving the Problem]

The present invention is a system for analyzing the number of visitors that addresses the issues raised above by identifying estimated visitors based on the IP address, Web browser type and the operating system under which the browser is running, and, further, by identifying persons engaged in continuous access for a set period of time as the same person.

[0010]

In addition, site administrators can embed specific META tags in Web pages or establish benchmark pages so that a robot search engine in the log analysis system can find benchmark pages and, with those pages as pivot points, observe page transitions before and after, thereby measuring how many visitors have browsed to specific pages. Also, by combining these means, log data from a plurality of servers can be efficiently analyzed.

[0011]

In other words, the log analysis system according to the present invention obtains an access log for a prescribed Web server and dynamically analyzes access trends to the Web server by users based on that log file, with the number of visitors estimated on the basis of IP addresses, identity of Web browser and operating system type, and the time interval between HTML file session transitions.

[0012]

In this case, if the time interval between session transitions is 30 minutes or less, a determination is made that the same visitor has accessed the server. Also, version information is provided about the Web browser and operating system type.

[0013]

In addition, the log analysis system according to the present invention dynamically analyzes access trends to a Web server by users by specifying a specific HTML file, locating that specific HTML file based on the log file using robot search means, and analyzing the chronological transitions between the HTML files before and after the specific HTML file.

[0014]

Also, the specific HTML file is specified by embedding a specific META tag in the header section of the specific HTML file in advance and locating the META tag based on the log file using robot search means.

[0015]

Furthermore, logs having a specific pattern of chronological transitions between the HTML files and logs partially matching with a specific pattern for chronological transitions between the HTML files are aggregated.

[0016]

Moreover, in the log analysis system according to the present invention, a plurality of prescribed Web servers exists, and the access logs for the plurality of Web servers are converted to a common format and transferred to a log database on a log analysis site.

[0017]

In this instance, there are cases in which the plurality of Web servers is composed of servers of different types, and server names along with access logs are transferred to the log database, and there are cases in which the plurality of Web servers is composed of servers of the same type, and only access logs are transferred to the log database.

[0018]

Finally, the log analysis system according to the present invention prepares a report showing the number of accesses, the number of visitors, and the access time on the basis of the estimated number of visitors, a report showing parameter-specific rankings of any parameter based on the number of accesses, the number of visitors, and the access time, and a report also showing the title, obtained by robot search means, in the header section of each analyzed HTML file.

[0019]

[Embodiments of the Invention]

The following section describes a preferred embodiment of the present invention in detail, with reference to drawings.

[0020]

One embodiment of the present invention is as follows. Fig. 1 shows an overall view of a network used with the present invention. In Fig. 1, 1 is the Internet, and the WWW server 10 for the client site is connected to a log analysis site via a router 2. To the log analysis site are connected a report distribution mail server 3, a WWW server 4, a data processing server 5, a log analysis server 6, and robot search server 7, a log database 8, a location database 9, and a client database 15. As described above, the log analysis site functions by accessing raw log databases 11 at individual client sites. These servers are equipped with generally known operating systems under which the program for the log analysis system runs.

[0021]

Fig. 2 is a flowchart showing the overall process flow of the log analysis site in the log analysis system according to the present invention. When the log analysis site starts operation (for example, once per day), client site access logs are collected by a log collection server 12 (S11). Next, in S12, data collection begins using a robot search server 7 to determine which files showing which Web pages at the client to target for analysis. In S13, the log collection server 12 converts the access logs into a common format (Fig. 10) for analysis by the log analysis server 6. This log format conversion is necessitated by the fact that three main types of log file formats are in use (Netscape™, NCSA, and IIS). As an example, in Fig. 9 a process for standardizing time formats is disclosed. In S14, access log analysis is started. In S15, visitor calculation is first performed, and in S16, if the process terminates with only a visitor estimate, control passes to S21. In S16, in a case in which a Web page transition ranking analysis is to be performed, in S17 the transition ranking analysis is performed, and control passes to S18. In S18, an instruction is issued whether or not to perform analysis on other parameters. If so, control passes to S19, and if not, control passes to S21. In S19, analysis is performed on other parameters, and then, in S21, an analysis report is automatically sent to each client, and the overall analysis is terminated. These results are all stored in a prescribed format, and the format shown in Fig. 14 is provided. In this case, a generally known database format is used to store the data.

[0022]

Fig. 3 shows the detailed steps from log file collection up to analysis process execution. In S30, when the program is launched, in S31 the client site is accessed, and control shifts to S32. In S32, the raw log database 11 stored on the client site is accessed. At the same time when the database is accessed, in S33, a determination is made how to acquire the raw log file via FTP (File Transfer Protocol, as used on the Internet). In a case in which the FTP command GET is used to acquire raw logs from the log analysis site, control shifts to S34, and after confirming whether or not the log file exists, in S35, a determination is made whether or not a log daily log file exists. If not, control automatically returns to S34 at a set time, and the process is re-executed. If a daily log file does exist, control passes to S36, and the log file is acquired. Here, if the FTP command PUT is used to transfer the file at a set time from the client site as the log acquisition method, control passes as-is to log file acquisition in S36. Thereafter, in S37, the log file is converted to the common format (Fig. 10), an ID is assigned to each individual file, and

the files are saved in the database (S38). After the file save process is complete, the process terminates (S39).

[0023]

Fig. 4 shows how the data saved in step S38 in Fig. 3 is analyzed. In S41, when the process starts, in S42 data sorting is performed. Here, the databases saved by ID (record) are sorted by source address, browser used (including version information), access date and time, and URL. Control then shifts to the visitor calculation step. In S44, first a determination is made whether or not the source addressees for the preceding and following records are the same (see the table in Fig. 10). If so, a determination is made whether or not the browser used with the preceding and following records is the same (S45). If so, a determination is made whether or not the access dates and times for the preceding and following records are within 30 minutes of one another (S46). If S44, S45 and S46 are all the same, a determination is made that the same visitor is involved, and control shifts to S49. If any one of S44, S45 or S46 is not the same, then the determination is recorded as a different visitor (S48). Then S48 and S49 are combined, and the visitor calculation terminates (S49). The determination time of 30 minutes for identifying a visitor as the same can fluctuate according to the nature of the page, and the time may be set to 15-45 minutes as appropriate. It is preferable to collect separate statistics on how to optimally set this time.

[0024]

By repeating this process record-by-record, it is possible to calculate the number of visitors, as shown in Fig. 11. At this time, logs are managed by assigning an ID to each visitor in the log database 8.

[0025]

Next, in Fig. 5, a search method in the robot search server for files of transition target Web pages is described. Once data collection begins (S58), the robot starts access to the top-level directory of each site (S59). Next, in S60, data collection is performed from the file source (material written in HTML source code), and in S61 a determination is made whether or not a title is declared. Here, if a title can be obtained, control shifts to S62, and the result is stored in the

location database 9. If no title tag is present, control shifts to S63. Next, in S63, a determination is made whether or not a META tag is listed in the source code of the accessed file. The listing method for the META tag stored in advance on the client side and used for specified pages is shown below.

[0026]

Meta tags are listed as follows:

- For the first page recording a transition:

```
<META NAME="flog-anchor" CONTENT="trans-top">
```

- For the last page recording a transition:

```
<META NAME="flog-anchor" CONTENT="trans-end">
```

[0027]

Here, if a META tag as shown above and in Fig. 13 is listed in the source, the data is stored in the location database 9 as a specified page for transition ranking (S64). Thereafter, in S65, a determination is made whether or not link data is present in the source. If so, control shifts to S66, execution moves to the link destination page, and the steps continue from S72. In S65, if no link data is present in the source, control shifts as-is to S67, and the robot search terminates.

[0028]

In Fig. 6, Web page transition rankings are calculated on the basis of the estimated number of visitors derived in Fig. 4 and on the location of transition target Web pages obtained in Fig. 5. Here, once the ranking calculation starts (S70), estimated visitor data is obtained (S71) and visitor ID specific analysis begins (S72). In S73, a determination is made for each visitor ID whether or not data is present that includes specified pages collected in Fig. 5. If so, the data is saved in the log database 8 as transition ranking target data. In S73, if no target data is present, then in S75 a determination is made whether or not data for the next visitor ID is present. If so, control returns to S72. In S75, if there are no records being analyzed, then in S76 the number of duplications in the data stored in the database for each pattern. In S77, rankings are displayed in a prescribed order as shown in Fig. 12, and the transition ranking calculation process terminates (S78).

[0029]

There are two types of transition rankings:

- 1) Rankings of transitions from specified pages, and
- 2) Rankings of transitions to specified pages

1) tracks to which pages transitions occurred from the specified page, while 2) tracks from which pages transitions occurred to the specified page. In addition, in the embodiment shown in Fig. 12, there are three tiers of transition page numbers, showing output ranking numbers three pages ahead from the specified page.

[0030]

In addition, Fig. 7 presents a flowchart of the process of analyzing other parameters. Here, it is possible to calculate the following rankings using the method employed in Fig. 6.

[0031]

- 1 No. of total accesses / no. of total visitors / total access time
- 2 No. of accesses / no. of visitors / access time by day of week
- 3 Access ranking by time frame
- 4 Access ranking by content
- 5 Access ranking by directory
- 6 Access ranking by subdomain
- 7 Access ranking by full domain
- 8 Access ranking by browser
- 9 Access ranking by operating system
- 10 Ranking by first page
- 11 Ranking by last page
- 12 Ranking by previous page (file)
- 13 Ranking by previous page (full domain)
- 14 Ranking by search keyword
- 15 Ranking by search engine

- 16 File transfer volume
- 17 Error log ranking
- 18 Visitor ranking by time frame
- 19 Visitor ranking by content
- 20 Visitor ranking by directory
- 21 Visitor ranking by subdomain
- 22 Visitor ranking by full domain
- 23 Visitor ranking by browser
- 24 Visitor ranking by operating system
- 25 Visitor ranking by first page
- 26 Total access time
- 27 Access time ranking by content
- 28 Access time ranking by directory
- 29 Access time ranking by subdomain
- 30 Access time ranking by full domain
- 31 Access time ranking by browser

In addition, Fig. 8 describes the log analysis process when a plurality of client sites is involved. Here, in S90, once the process starts, in S91 log file acquisition begins. In S92, a determination is made whether reports are subject to merge or mirror processing. "Merge" refers to a case in which a plurality of logs is to be aggregated into the same report, and the logs merely need to be consolidated. This process is used, for example, in a case in which an information provision server is separated from a payment processing server in online shopping. "Mirror" refers to a case in which a plurality of logs targeted for aggregation for the same report exists but the logs are regarded as the same for aggregation. This case refers to a situation in which an online shopping server on which the same content is stored on a plurality of machines in order to distribute the processing load is connected at the same address. Next, in S92, in a case in which processing is performed, in S93 server information is checked, and in S94 analysis processes are performed on the respective log files obtained. In S95, a determination is made whether to perform the merge process or the mirror process. If the merge process is chosen, control shifts to S96, while if the mirror process is chosen, control shifts to S98. In S96, the respective analysis

processes are output as separate items in the same report, and control shifts to S97. In S98, the respective analysis processes are viewed identically for output in the same report, and control shifts to S97. In addition, in S92, if no processing is to be performed for a target report, control shifts to S99, where the obtained file is analyzed and processed, and control shifts to standard output in S100. In S97, after the output process is complete, this process terminates.

[0032]

Also, in the present embodiment, a method is disclosed involving use of a robot to search for META tags. However, the invention is not restricted to this method. It is possible to specify a particular page from a client site or from the log analysis system and analyze it using a similar method without using META tags.

[0033]

[Effect of the Invention]

According to the present invention, it is possible to eliminate the disadvantage of being unable to figure out the number of Web page visitors using a calculation based on IP addresses or browser cookies, as in the past, because of being unable to obtain the data. It is possible to estimate the number of visitors based on the IP address, the type of Web browser, and the operating system under which the browser runs, and thereby more accurately estimate the number of visitors.

[0034]

In addition, since visitors are identified as the same based on the time interval between page transitions, errors of tabulating accesses by the same visitor as accesses by a plurality of visitors are reduced, and errors of tabulating other accesses by other persons as accesses by the same visitor are reduced.

[0035]

Furthermore, since it is possible to easily search for specified files using a robot search system and use that information along with estimated visitor numbers to track trends in visitor numbers and page transitions -- that is, Web page visitors -- without analyzing all log files as in the past, resources are saved, and system usability is improved. In addition, it is possible to smoothly

perform consolidated statistical processing on a plurality of logs, which was impossible using package software products heretofore, thus reducing costs and the number of process steps.

[Brief Description of Drawings]

Fig. 1 Overall view of a network employing the present invention.

Fig. 2 View of the overall operation flow of the log analysis system according to the present invention.

Fig. 3 Flowchart showing the detailed process steps from log file acquisition up to analysis process execution.

Fig. 4 Flowchart showing the visitor number (estimated visitor) calculation process.

Fig. 5 Flowchart showing the data aggregation process flow using the robot search engine.

Fig. 6 Flowchart showing the calculation process for transition rankings using the number of estimated visitors obtained in Fig. 4.

Fig. 7 Flowchart showing analysis of other parameters.

Fig. 8 Flowchart showing an example analysis using logs from a plurality of servers.

Fig. 9 View showing intermediate file transitions.

Fig. 10 View showing an example of an intermediate file converted to a common format.

Fig. 11 View showing an output sample of a log analysis report.

Fig. 12 View showing a display showing transition rankings and an output sample.

Fig. 13 View showing an example of embedded META tags used for robot searching.

Fig. 14 View showing a database table used by this log analysis system.

[List of Symbols]

- | | |
|---|----------------------------------|
| 1 | Internet |
| 2 | Router |
| 3 | Report distribution email server |
| 4 | Web server |
| 5 | Data processing server |
| 6 | Log analysis server |
| 7 | Robot search server |

- 8 Log database
- 9 Location database
- 10 Web server (client)
- 11 Raw log
- 12 Log collection server
- 13 Firewall
- 14 LAN
- 15 Client database

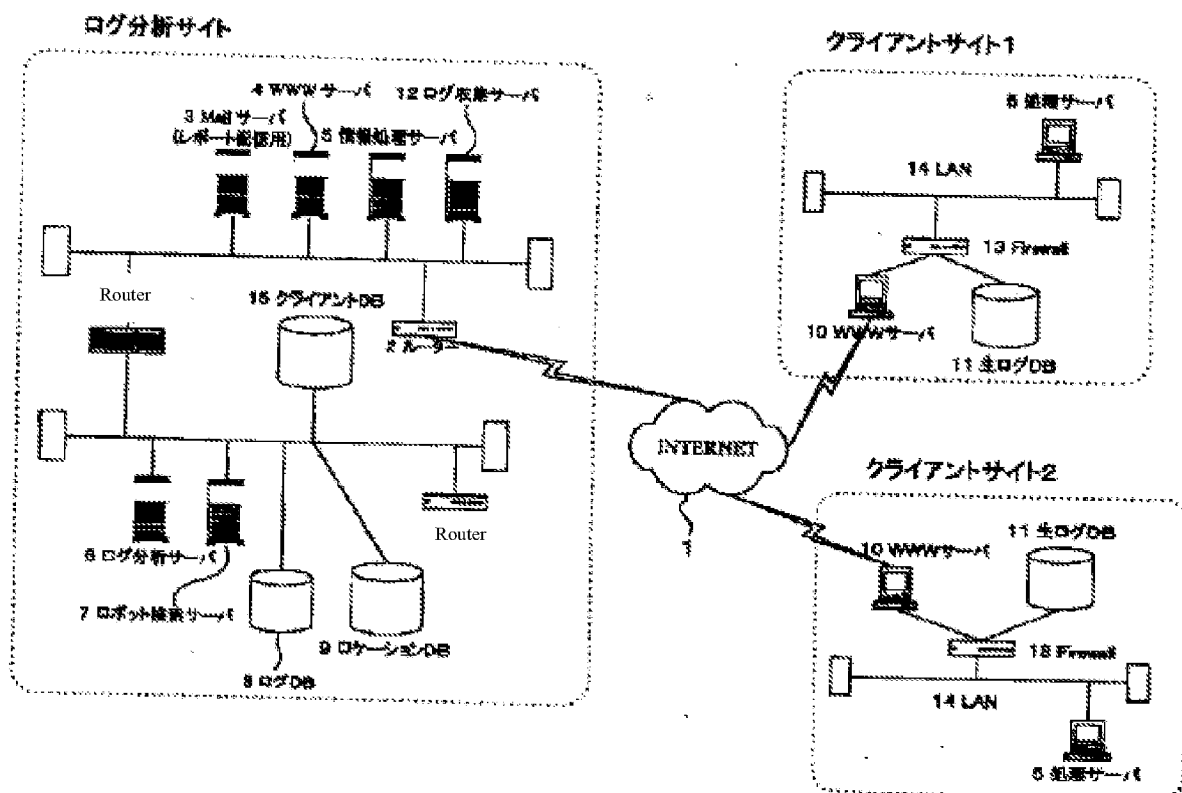
Fig. 13

--- Example -----

```
<html>
<head>
<title>Log Analysis Service</title>
<META NAME="flog-anchor" CONTENT="trans-top">
<meta http-equiv="Content-Type" content="text/html; charset=x-sjis">
</head>
```

<<== [ログ分析システムV1.1用TAG]

Fig. 1



{left-hand box}

Log Analysis Site

3 Mail Server (for report distribution)

4 Web server

5 Data processing server

12 Log collection server

15 Client database

2 Router
6 Log analysis server
7 Robot search server
8 Log database
9 Location database

{upper right box}
Client Site 1
5 Processing server
10 Web server
11 Raw log database

{lower right box}
Client Site 2
10 Web server
11 Raw log database
5 Processing server

Fig. 9

[Intermediate File 1]

At applicable lines in the input file, date and time data is recalculated as JST and output in the format [YYYY/MM/DD HH:MM:SS], with tabs replaced by single spaces.

Also, lines that begin with "#" or "format" are output as-is.

<example conversion>

(apache)
(omitted) kwgcc-01p52.ppp.odn.ad.jp - - [29/Jul/1999:00:00:00 +0900] "GET /POWERAXEL/image/s....

↓
kwgcc-01p52.ppp.odn.ad.jp - - [1999/07/29 00:00:00] "GET /POWERAXEL/image/s.... (N8)

(iis3)
(omitted) 202.219.168.66, -, 99/06/26, 0:00:00, W3SVC1, YBPASP2, 202.221.4.99, 5406, 247, 13....

↓
(omitted) 202.219.168.66, -, [1999/06/26 00:00:00], W3SVC1, YBPASP2, 202.221.4.99, 5406, 247, ..

(iis4) (In iis4, an assumption is made that GMT has been used. In addition, since the date is omitted, an assumption is made that 15:00:00 to 23:59:59 refers to the day before and 00:00:00 to 14:59:59 refers to the current day.)

(omitted)

↓
(omitted) [1999/07/26 09:00:02] 210.131.44.143 - GET /olv/form_q.asp - 200 19641 Mozilla/3.0+W..

(if netscape, time_diff=60)

(omitted) 210.131.71.98 - - [12/Jun/1999:07:19:10 +0900] "GET /uc_checkproc/ings/topcalcostb....

↓
210.131.71.98 - - [1999/07/12/ 08:19:10] "GET /uc_checkproc/ings/topcalcostb.... (R8)

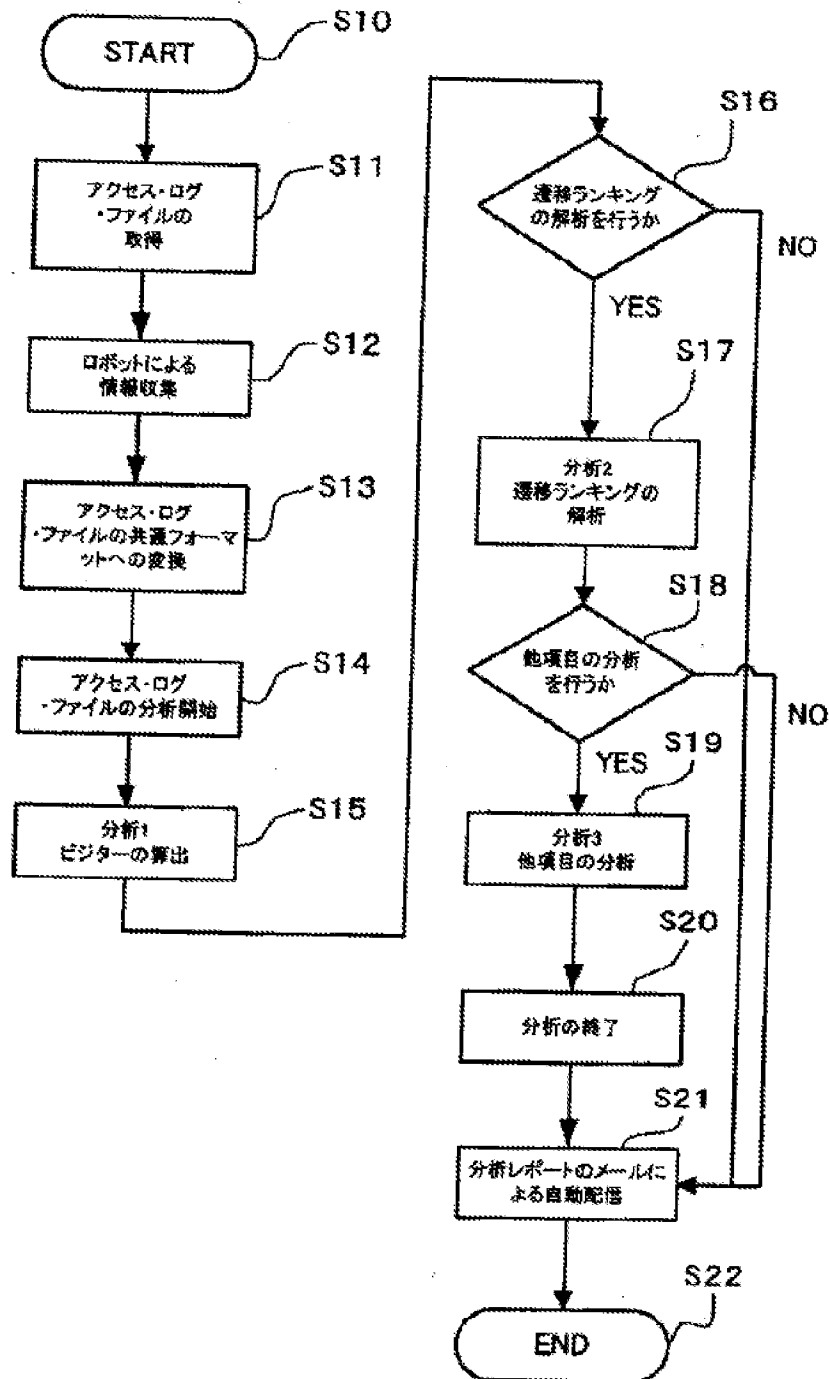
(if wu-ftpd, time_diff=60)

(omitted) Mon Aug 23 16:58:21 1999 935 202.211.205.166 3393630 /19990129/import/MAX/990823b.....

↓
(omitted) [1999/08/23 16:58:21] 935 202.211.205.166 3393630 /19990129/import/MAX/990823b..... (

Fig. 2

Log Analysis System Overall Flow



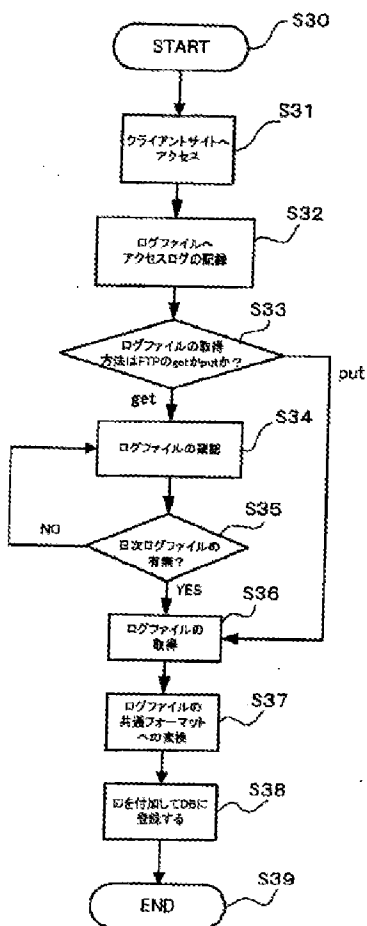
S11 Acquire access log files

S12 Use robot to collect data

- S13 Convert access log files to common format
- S14 Start access log file analysis
- S15 Analysis 1: Calculate no. of visitors
- S16 Analyze transition rankings?
- S17 Analysis 2: Analyze transition rankings
- S18 Analyze other parameters?
- S19 Analysis 3: Analyze other parameters
- S20 End analysis
- S21 Automatically distribute analysis reports by email

Fig. 3

Flow from Log File Acquisition to Analysis Process Execution



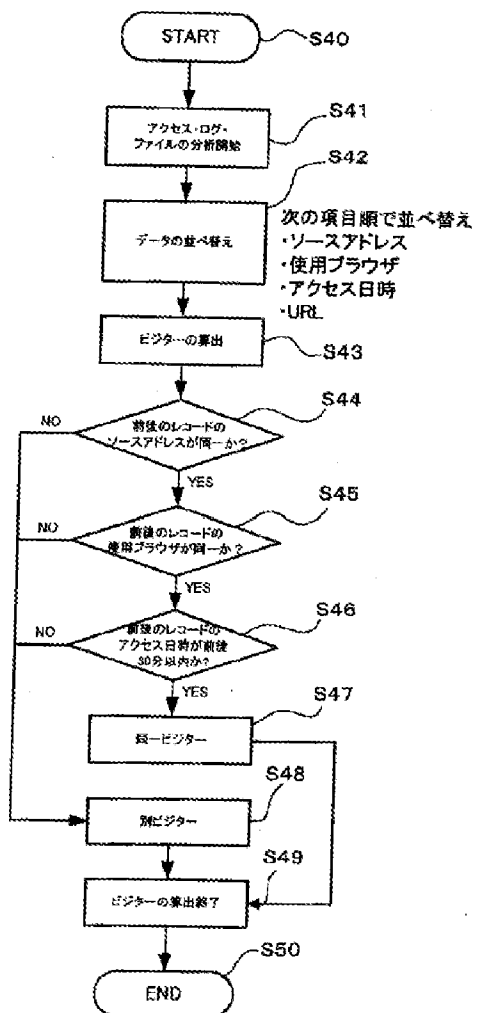
S31 Access client site

- S32 Save access log in log file
- S33 Log file acquisition method is FTP command Get or Put?
- S34 Check for log file
- S35 Daily log file available?
- S36 Acquire log file
- S37 Convert log file to common format
- S38 Assign ID and save in database

Fig. 4

Log Analysis Process Flow 1

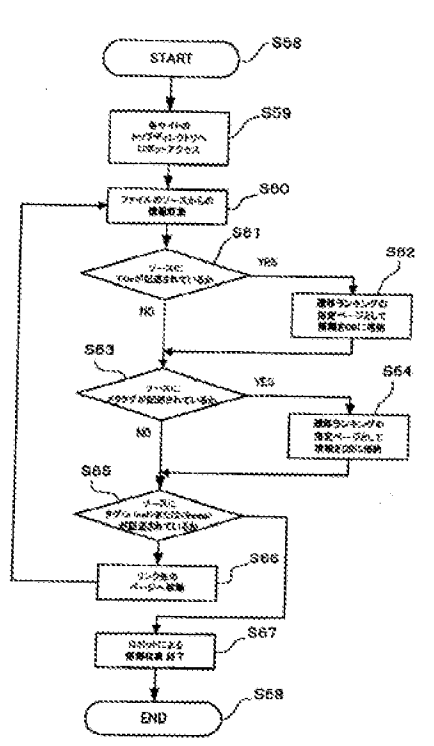
Visitor Calculation



- S41 Start access log file analysis
- S42 Sort data
- Sort by following parameters
- Source address
 - Browser used
 - Access date and time
 - URL
- S43 Calculate visitors
- S44 Source address same for previous and next records?
- S45 Browser same for previous and next records?
- S46 Access date and time within 30 minutes for previous and next records?
- S47 Same visitor
- S48 Different visitor
- S49 End visitor calculation

Fig. 5

Flow of Data Collection by Robot

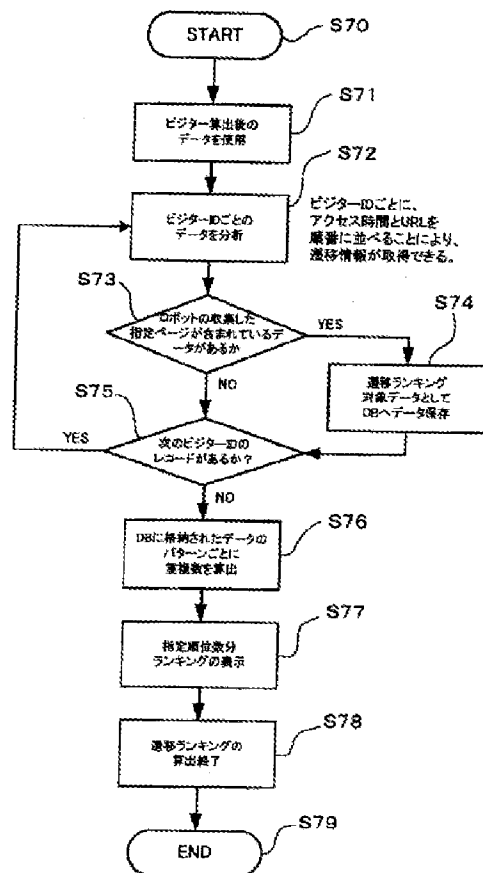


- S59 Robot accesses top-level directory of each site
- S60 Collect data from file source
- S61 Title listed in source?
- S62 Save data in database as transition ranking specified page
- S63 Meta tag listed in source?
- S64 Save data in database as transition ranking specified page
- S65 <a href>or <frematag> listed in source?
- S66 Go to link destination page
- S67 End data collection by robot

Fig. 6

Log Analysis Process Flow 1

Transition Ranking



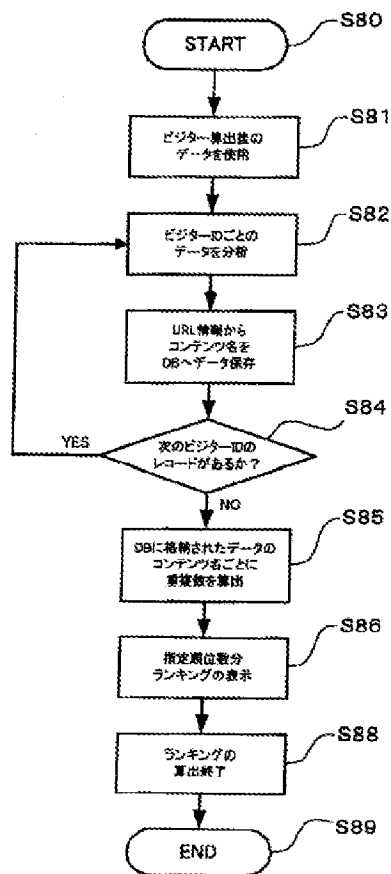
- S71 Use data post-visitor calculation
- S72 Analyze data by visitor ID
Sort each visitor ID by access time and URL to obtain transition data.
- S73 Data present containing specified pages collected by the robot?
- S74 Save data in database as transition ranking target data
- S75 Next visitor ID record present?
- S76 Calculate number of duplications for each pattern in the data stored in the database
- S77 Display rankings for the specified ranking numbers
- S78 End transition ranking calculation

Fig. 7

Log Analysis Process Flow

Example Analysis of Other Parameter

[Visitor Ranking by Content]

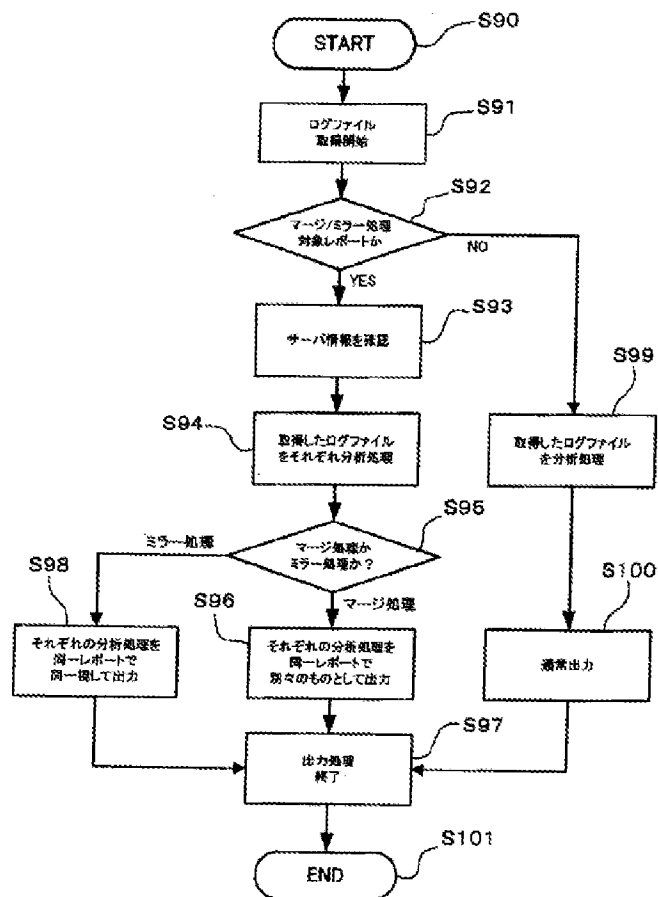


- S81 Use data post-visitor calculation
- S82 Analyze data by visitor ID
- S83 Save data on content name from URL data to database
- S84 Next visitor ID record present?
- S85 Calculate the number of duplications for each content name in the data stored in the database
- S86 Display rankings for the specified ranking numbers
- S88 End ranking calculation

Fig. 8

Log Analysis Process Flow

Merge/Mirror Processing for a Plurality of Log Files



- S91 Start log file acquisition
- S92 Any reports subject to merge/mirror processing?
- S93 Check server data
- S94 Analyze each log file obtained
- S95 Merge process or mirror process?
- Mirror process
- S98 Output the analysis processes treating all as the same for the same report
- S96 Output the analysis processes treating each as separate for the same report
- S97 End output process
- S99 Analyze log files obtained
- S100 Standard output

Fig. 10

[Intermediate File]

Data sample, converted to common format, sorted by the following keys:

- Source address
- Browser used
- Access date and time

*To make the listing more intelligible, blank lines are placed between records. These lines are not present in the actual file.

6.147

Access Time per Access (T[mins.]/A)	0.499
Access Time per Visitor (T[mins.]/V)	3.069

Actual Numbers by Day of Week

Day Accesses (A) No. of Visitors [People] (V) Access Time [time] (T)

Monday

Tuesday

Wednesday

Thursday

Friday

Saturday

Sunday

Coefficient Table by Day of Week

Day (A/V[people]) (T[mins.]/A) T[mins.]/V[people])

Monday

Tuesday

Wednesday

Thursday

Friday

Saturday

Sunday

No. of Accesses by Time Frame

Time Frame	No. of Accesses	No. of Accesses /No. of Visitors	Access Time [mins.] /No. of Accesses
------------	-----------------	-------------------------------------	---

Fig. 12

<Transition Rankings>

[Linear Rankings from Specified Page (TOP)]

[Top 3]

No. of Visitors

Content

[Specified Page 1] /xxx/campaign1.html [Campaign 1]

2852 /xxx/campaign1.html
 /index3.html
 /index4.html
 /index5.html
 /campaign/info.html

2000 /xxx/campaign1.html
 /index3.html
 /index4.html
 /index5.html
 /campaign/info2.html

1000 /xxx/campaign1.html
 /index2.html
 /index4.html
 /index5.html
 /info.html

[Specified Page 2] /xxx/campaign2.html

2823 /xxx/campaign2.html
 /test4.html
 /index4.html
 /index5.html
 /sample212.html

2000 /xxx/campaign2.html
 /index3.html
 /index4.html
 /index5.html
 /campaign/info2.html

1000 /xxx/campaign2.html
 /index2.html
 /info.html
 /index4.html
 /index5.html

Fig. 14

Customer Server Data Table

Customer ID
Report unit ID
Report name
Customer server ID
Customer server name
OS
FTP
FTP-ACCOUNT
Weekly
Monthly
Send destination address
Service start date
Service end date

Staff Table

Staff ID
Name
Department
e-mail
Web ID
Web password

Access Data Table

Access date
Access source FQDN
Access source subdomain
Access source full domain
Access destination URL
Referrer
Search keyword
Same-access flag

Service Provider Master

Service provider ID
Service provider name
FROM
Reply-to
Errors-to
Staff contact person

Robot Master Table

Report unit ID
Collection range
Application start date
Application end date
Protocol
Port no.
Content path
Data 1
Data 2
Data 3
Data 4
Data 5
Type
Entry date

Service Data Table by
Customer

Report unit ID
Analysis type
Cycle
Output rank
Max. no. of specified pages
Max. no. of transitions
Entry date